EECS 10: Computational Methods in Electrical and Computer Engineering
Lecture 1

Brian Demsky
bdemsky@uci.edu

The Henry Samueli School of Engineering
Electrical Engineering and Computer Science
University of California, Irvine
Lecture 1: Overview

- Introduction
- Course administration
  - Course web pages
- Getting started
  - Obtain your UCInetID
  - Obtain an account on the EECS servers
  - Log into the server
  - Work in the Unix system environment
  - Use a text editor
Introduction

- Course Contents
  - Introduction to structured programming
  - Binary data representation
  - Hands-on experience
    - High-level structured programming language
  - Introduction to algorithm efficiency
  - Applications of structured programming
  - Solving engineering problems
Course Administration

- Course web pages online at http://eee.uci.edu/06w/15300/
  - Instructor information
  - Course description and contents
  - Course policies and resources
  - Course schedule
  - Homework assignments
  - Course communication
    - Mailing list (announcements)
    - Email (administrative issues)
Getting Started

• Obtain your UCInetID
  • Your unique ID at UCI
  • Activation online at NACS web pages: http://activate.uci.edu/activate/menu.html

• Obtain an account on the EECS servers
  • Your working account in EECS
  • TA’s will have your login information shortly
Getting Started

• Log into the server
  • Terminal with SSH protocol (secure shell)
  • EECS servers
    • east.eecs.uci.edu
    • newport.eecs.uci.edu
    • malibu.eecs.uci.edu
  • User name, password
• Work in the Unix system environment
  • shell, command prompt
  • system commands
    echo, date, ls, cat, man, more,
    pwd, mkdir, cd, cp, mv, rm, rmdir
  • manual pages
  • compiling C code
Unix Commands

- Unix system commands
  - echo  print a message
  - date  print the current date and time
  - ls    list the contents of the current directory
  - cat   list the contents of files
  - more  list the contents of files page by page
  - pwd   print the path to the current working directory
  - mkdir create a new directory
  - cd    change the current directory
  - cp    copy a file
  - mv    rename and/or move a file
  - rm    remove (delete) a file
  - rmdir remove (delete) a directory
  - man   view manual pages for system commands
Text Editors

- Use a text editor
  - pico  (easy-to-use editor)
  - emacs  (powerful editor)
  - others...
- Edit on PC and upload to UNIX system
To translate C code into an executable, we use a compiler.

We will use gcc in this course.

To translate file.c into an executable, we run:

```
gcc file.c
```

The compiler reads file `file.c` and creates file `a.out`.

Options may be specified to direct the compilation:

- `-o file` specifies output file name.
- `-ansi` specifies ANSI code.
- `-Wall` turns on all warnings.
Alternate Programming Environments

• You can use any platform you wish to write course assignments
• You can install gcc on your own machine (MS Windows, Macintosh, Linux)
• You can use any text editor to write your code in
• But check that your assignments run on the Sun machines before turning them in
• You have to use Sun machines to turn in your assignments